



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/887,622	06/22/2001	Cary Lee Bates	ROC920010090US1	7424

7590 06/25/2004
Gero G. McClellan
Thomason, Moser & Patterson, L.L.P.
3040 Post Oak Boulevard, Suite 1500
Houston, TX 77056-6582

EXAMINER

PHAM, CHRYSTINE

ART UNIT PAPER NUMBER

2122

DATE MAILED: 06/25/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/887,622	Applicant(s) BATES ET AL.	
	Examiner Chrystine Pham	Art Unit 2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☐ Responsive to communication(s) filed on ____.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-42 is/are pending in the application.
- 4a) Of the above claim(s) ____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) ____ is/are allowed.
- 6) ☒ Claim(s) 1-42 is/are rejected.
- 7) ☐ Claim(s) ____ is/are objected to.
- 8) ☐ Claim(s) ____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 22 June 2001 is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. ____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. ____. |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date ____. | 6) <input type="checkbox"/> Other: ____. |

DETAILED ACTION

Drawings

1. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because:
 - o They include the following reference character(s) not mentioned in the description:
206 (FIG.2).
 - o They do not include the following reference character(s) mentioned in the description:
bus 114 (of FIG.1, mentioned in [0025] line 3 & [0026] line 6), 205 (of FIG.2, mentioned in [0034] line 2).
2. The drawings are objected to under 37 CFR 1.83(a) because they fail to show 137, 138, 140, 142 (reference characters of FIG.1) and 408, 410 (reference characters of FIG.4) as described in the specification.
 - o Per reference characters 137, 138, 140 & 142 of FIG.1, line 2-4 of [0024] states " ... by a mass storage interface 137 operably connected to a direct access storage device 138, by a video interface 140 operably connected to a display 142". However, there is no clear indication of the **connection** between 137 and 138 or that of 140 and 142.
 - o Per 408 & 410 of FIG.4, line 15-16 of [0042] states " ... further divided into a live variables pane 408 and a dead variables pan 410". However, 408 labels Dead Variables and 410 labels Live Variables in FIG.4.

Any structural detail that is essential for a proper understanding of the disclosed invention should be shown in the drawing. MPEP § 608.02(d).

3. Corrected drawing sheets are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is

Art Unit: 2122

being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. The replacement sheet(s) should be labeled "Replacement Sheet" in the page header (as per 37 CFR 1.84(c)) so as not to obstruct any portion of the drawing figures. If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Specification

4. The disclosure is objected to because of the following informalities: inconsistency in term "CFG 220" (see [0036] line 2) since CFG has been previously referenced as 154 (see [0035] line 1 & 154 of FIG.1) and 220 has been previously referring to start node (see [0035] line 1 & 220 of FIG.2). Appropriate correction is required.

Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

*A person shall be entitled to a patent unless –
(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.*

6. Claims 1-8, 10-15, 17-24, 27, 29-30, 32-34, and 37-41 are rejected under 35 U.S.C. 102(b) as being anticipated by Sumi et al. (U.S. Patent 5,881,288) (hereinafter *Sumi et al.*).

Art Unit: 2122

As per claim 1, *Sumi et al.* disclose a computer system, comprising an output device (e.g., FIG.4 502 & associated text) and at least one processor (e.g., FIG.7 & associated text) which, when executing a debugging program, is configured to:

- wait for a program being debugged to stop executing immediately prior to executing a next executable statement at which at least one variable has a current value (e.g., col.24 : 23-27 & col.27 : 48-52); and
- display on the output device the at least one variable in a manner that visually indicates an executable status of the at least one variable, wherein the executable status is indicative of at least one of a use and change of the current value during subsequent continuing execution of the program being debugged (e.g., col.23 : 51-60).

As per claim 2, *Sumi et al.* disclose a system as applied to claim 1, wherein the executable status indicates that the current value may change when the next executable statement is executed (e.g., col.23 : 31-36).

As per claim 3, *Sumi et al.* disclose a system as applied to claim 1, wherein the executable status indicates that the current value may be used when the next executable statement is executed (e.g., col.27 : 39-41).

As per claim 4, *Sumi et al.* disclose a system as applied to claim 1, wherein the executable status is visually represented on the output device to differentiate the at least one variable from other variables displayed on the output device (e.g., FIG.9A *operation-possible variable display window* & FIG.9B).

As per claim 5, *Sumi et al.* disclose a system as applied to claim 1, further comprising a memory containing a monitor window interface (e.g., FIG.4 214) configured to display the at least one variable on the output device in a manner to

Art Unit: 2122

visually differentiate the at least one variable from other variables having a different executable status (e.g., FIG. 9A *operation-possible variable display window* & FIG. 9A & associated text).

As per claim 6, *Sumi et al.* disclose a method for displaying variables of a program being debugged, comprising:

- when the program being debugged stops executing immediately prior to executing a next executable statement at which at least one variable has a current value (e.g., col.24 : 23-27 & col.27 : 48-52), determining at least one of a first executable status and a second executable status of the at least one variable based on a current point of execution, wherein the first executable status is defined by whether the current value of the at least one variable may change during subsequent execution of the program being debugged and the second executable status is defined by whether the current value of the at least one variable has a use during subsequent execution of the program being debugged (e.g., FIG.18A S76 & S79, col.6 : 56-62, col. 11 : 47-52, col.15 : 27-36, col.18 : 25-35, col.23 : 25-41 and 51-60, and col.27 : 30-36); and
- preparing an output which, when displayed on an output device, visually indicates an executable status of the at least one variable at the current point of execution (e.g., FIG.6A & FIG.5D *primitive storage unit* & associated text, FIG.9A, and col.15 : 32-49).

As per claim 7, *Sumi et al.* disclose a method as applied to claim 6, wherein the second executable status is defined according to one of only the next executable statement and any statement that may be encountered during subsequent execution of the program being debugged (e.g., col.18 : 26-35).

Art Unit: 2122

As per claim 8, *Sumi et al.* disclose a method as applied to claim 6, wherein the program being debugged stops executing upon encountering a breakpoint (e.g., col.19 : 25-42, and col.21 : 38-40).

As per claim 10, *Sumi et al.* disclose a method of claim 6, wherein preparing comprises preparing the output so that, when displayed, the at least one variable is visually differentiable from other displayed variables according to their respective executable statuses (e.g., FIG.6A & FIG.5D *primitive storage unit* & associated text, FIG.9A, and col.15 : 32-49).

As per claim 11, *Sumi et al.* disclose a method of claim 6, wherein the first executable status indicates that the value of the at least one variable may change during subsequent execution of the program being debugged (e.g., FIG.12 S43-S49 & associated text).

As per claim 12, *Sumi et al.* disclose a method of claim 6, wherein the first executable status indicates that the value of the at least one variable may change during subsequent execution of the program being debugged and the second executable status indicates that the value of the at least one variable has a use during subsequent execution of the program being debugged (e.g., FIG.12 S43-S49 & associated text, col.27 : 30-43).

As per claim 13, *Sumi et al.* disclose a method of claim 6, wherein at least one variable is a variable referenced in the next executable statement in the program being debugged (e.g., col.15 : 26-28, FIG.9A variable *x* on line 10 & 11 of *line information display window*, *line display window*, and *operation-possible variable display window*).

As per claim 14, *Sumi et al.* disclose a method of claim 13, wherein the next executable statement contains a plurality of variables (e.g., FIG.12 S48 & S49 & associated text), and wherein preparing comprises preparing the output so that, when displayed, the at least one variable and the plurality of variables are visually differentiable from one another according to their respective different executable statuses (e.g., FIG.6A & FIG.5D *primitive storage unit* & associated text, FIG.9A, and col.15 : 32-49, col.23 : 25-32).

As per claim 15, it recites limitation which has been addressed in claim 14 above, therefore, is rejected for the same reason as cited in claim 14.

As per claim 17, *Sumi et al.* disclose a method for displaying variables of a program being debugged, comprising:

- when a program being debugged stops executing immediately prior to a next executable statement at which at least one variable has a current value (e.g., col.24 : 23-25, and col.27 : 48-52), determining an executable status of at least one variable of the statement based on a current point of execution, wherein the executable status is indicative of at least one of a possible use and a possible change of the current value during subsequent continuing execution of the program being debugged (e.g., FIG.18A S76 & S79, col.6 : 56-62, col. 11 : 47-52, col.15 : 27-36, col.18 : 25-35, col.23 : 25-41 and 51-60, and col.27 : 30-36); and
- preparing an output which, when displayed on an output device, visually indicates the executable status of the at least one variable at the current point of execution (e.g., FIG.6A & FIG.5D *primitive storage unit* & associated text, FIG.9A, and col.15 : 32-49, col.23 : 25-32).

Art Unit: 2122

As per claim 18, *Sumi et al.* disclose a method of claim 17, wherein the executable status is defined according to one of only the next executable statement and any statement that may be encountered during the subsequent continuing execution of the program being debugged (e.g., FIG.12 S43-S49 & associated text, col.18 : 26-34, and col.24 :5-8).

As per claim 19, *Sumi et al.* disclose a method of claim 17, wherein the program being debugged stops executing upon encountering a breakpoint (e.g., col.19 : 25-42, and col.21 : 38-40).

As per claims 20-22, they recite limitations which have been addressed in the above claims 12, 10, and 14 (respectively), therefore, are rejected for the same reasons as cited in claims 12,10, and 14 respectively.

As per claims 23-24, they recite limitations which have been addressed in both of the above claims 12 & 10, therefore, are rejected for the same reasons as cited in both claims 12 & 10.

As per claim 27, it recites limitation which has been addressed in the above claim 10, therefore, is rejected for the same reason as cited in claim 10.

As per claim 29, *Sumi et al.* disclose a signal bearing medium (e.g., FIG.4), comprising a debugging program which, when executed by a processor, performs a method, comprising:

- o when a program being debugged stops executing immediately prior to a next executable statement at which at least one variable has a current value (e.g., col.24 : 23-27 & col.27 : 48-52), determining an executable status of at least one variable of the statement based on a current point of execution, wherein the

Art Unit: 2122

executable status is indicative of at least one of a possible use and a possible change of the current value during subsequent continuing execution of the program being debugged (e.g., FIG.18A S76 & S79, col.6 : 56-62, col. 11 : 47-52, col.15 : 27-36, col.18 : 25-35, col.23 : 25-41 and 51-60, and col.27 : 30-36); and

- o preparing an output which, when displayed on an output device, visually indicates the executable status of the at least one variable at the current point of execution (e.g., FIG.6A & FIG.5D *primitive storage unit* & associated text, FIG.9A, and col.15 : 32-49).

As per claims 30 and 32, they recite limitations which have been addressed in the above claims 8, and 7 respectively, therefore, are rejected for the same reasons as cited in claims 8, and 7.

As per claims 33-34, 37-39, and 41, they recite limitations which have been addressed in the above claims 10, and 12, therefore, are rejected for the same reasons as cited in claims 10 & 12.

As per claim 40, it recites limitations which have been addressed in the above claim 22, therefore, is rejected for the same reasons as cited in claim 22.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 2122

8. Claims 9, 25-26, 31, 35-36, and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Sumi et al.* as applied to claims 6, 17, and 29 above, and further in view of Miyata et al. (U.S. Patent 5,165,036) (hereinafter *Miyata et al.*).

As per claim 9, *Sumi et al.* teach a method as applied to the above claim 6. *Sumi et al.* fail to teach determining which comprises referring to a control flow graph (CFG). However, *Miyata et al.* disclose a method for displaying variables of a program being debugged, wherein determining comprises referring to a CFG (e.g., FIG.18, FIG.20, see *data flow program* col.4 : 22-27). It would have been obvious that one of ordinary skill in the pertinent art at the time of applicant's invention would be motivated to modify the teaching of *Sumi et al.* to include the use of a CFG as disclosed by *Miyata et al.*, since information contained in the CFG can be used by the debugging program to automatically identify the decision points in the debugged program, whereat breakpoints are set, automatically.

As per claim 25, *Sumi et al.* teaching as modified by *Miyata et al.* (see above claim 9) teach a method as applied to claim 17, wherein determining comprises accessing a variable-containing data structure associated with the next executable statement (e.g., FIG.4 *function information storage unit 1042* & associated text, col.23 : 42-61).

As per claims 26, 31, and 35, they recite limitations which have been addressed in the above claim 25, therefore, are rejected for the same reasons as cited in claim 25.

As per claim 36, *Sumi et al.* teaching as modified by *Miyata et al.* (see above claim 9) teach the signal bearing medium of claim 35, wherein preparing

Art Unit: 2122

comprises preparing the output so that, when displayed, the variables are visually differentiable from one another according to their respective executable statuses (see above claim 10).

9. Claims 16, 28, and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Sumi et al.* as applied to claims 6, 17, and 29 (respectively) above, and further in view of *Bates et al.* (U.S. Patent 6,658,649) (hereinafter *Bates et al.*).

As per claim 16, *Sumi et al.* teach a method as applied to above claim 15. *Sumi et al.* fails to teach the formatting comprises at least one of brackets, parentheses, asterisks, highlighting, strike-outs and numerals. However, *Bates et al.* disclose a method for displaying variables (e.g., col.5 : 27-30), a step region which can be formatted by highlighting, shading, coloring, and the like (e.g., col.7 : 23-29), and the executed statements contained within the step region which can be formatted using asterisks, highlighting, etc. (e.g., col.7 : 43-51). Even though, *Bates et al.* are not explicit, in particular, on the formatting of variables, it would have been obvious that the teaching of *Bates et al.* could have been easily modified to include the formatting of variables which comprise at least one of brackets, parentheses, asterisks, highlighting, strike-outs and numerals. In addition, it would have been obvious to one of ordinary skill in the pertinent art at the time of applicant's invention to modify the teaching of *Sumi et al.* to include the display formatting as disclosed by *Bates et al.*, since such display formats would further enhance visual distinctness of variables and further support their identification during debugging.

As per claims 28 and 42, they recite limitations which have been addressed in the above claim 16, therefore, are rejected for the same reasons as cited in claim 16.

Conclusion

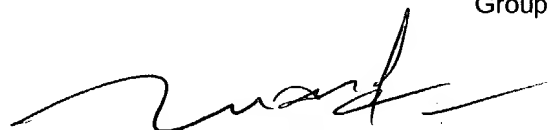
10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. ***.
 - o Method for determining the status of variables during the execution of optimized code, Kesselman et al. (US-6,678,884).
 - o Method and apparatus for debugging of optimized code, Olsen et al. (US-6,263,489)
 - o Method and apparatus for debugging of optimized code using emulation, Mirani et al. (US-6,434,741)
 - o Dynamic object visualization and browsing system , West, Alan A. (US-5,740,440)
 - o Collection of timing and coverage data through a debugging interface, Morshed et al. (US-6,721,941)
 - o Debugging a computer program by simulating execution forwards and backwards in a main history log and alternative history logs, Bishop et al. (US-5,784,552)
 - o Data processing system having monitoring of software Activity, King Michael Roy (US-5,983,366)
 - o Integrated dynamic-visual parallel debugging apparatus and method thereof, On et al. (US-6,275,956)
 - o System and method for automated testing and monitoring of software applications, Wild et al. (US-5,671,351)
 - o Method and apparatus for user side scheduling in a multiprocessor operating system program that implements distributive scheduling of processes, Spix et al. (US-6,195,676)
11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chrystine Pham whose telephone number is 703.605.1219. The examiner can normally be reached on Mon-Fri, 8:30am-5pm.

Art Unit: 2122

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q Dam can be reached on 703.305.4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

12. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Chrystine Pham
Examiner
Group Art Unit 2122



TUAN DAM
SUPERVISORY PATENT EXAMINER